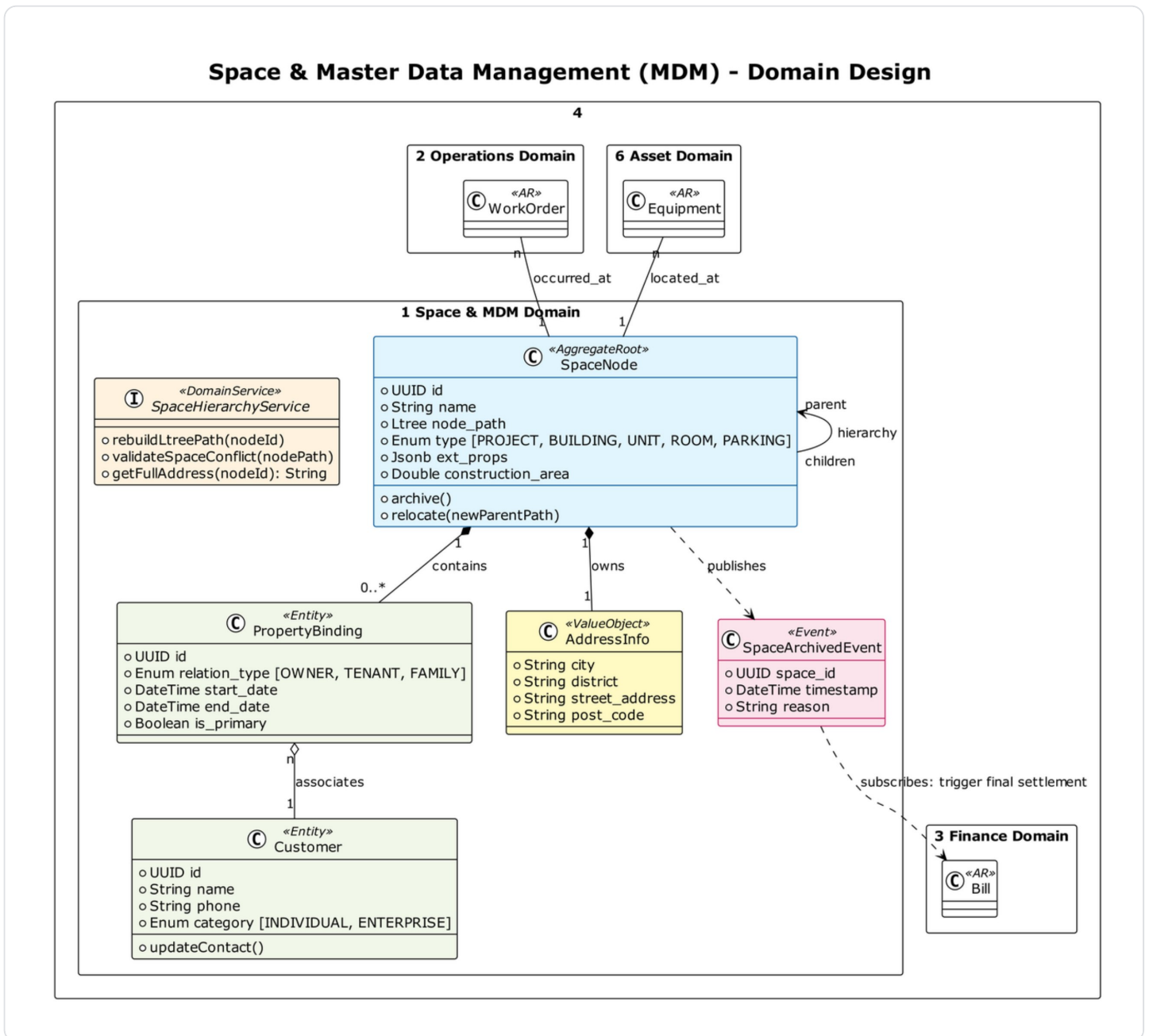


领域模型详细设计

1. 空间与主数据领域DDD

UML图：



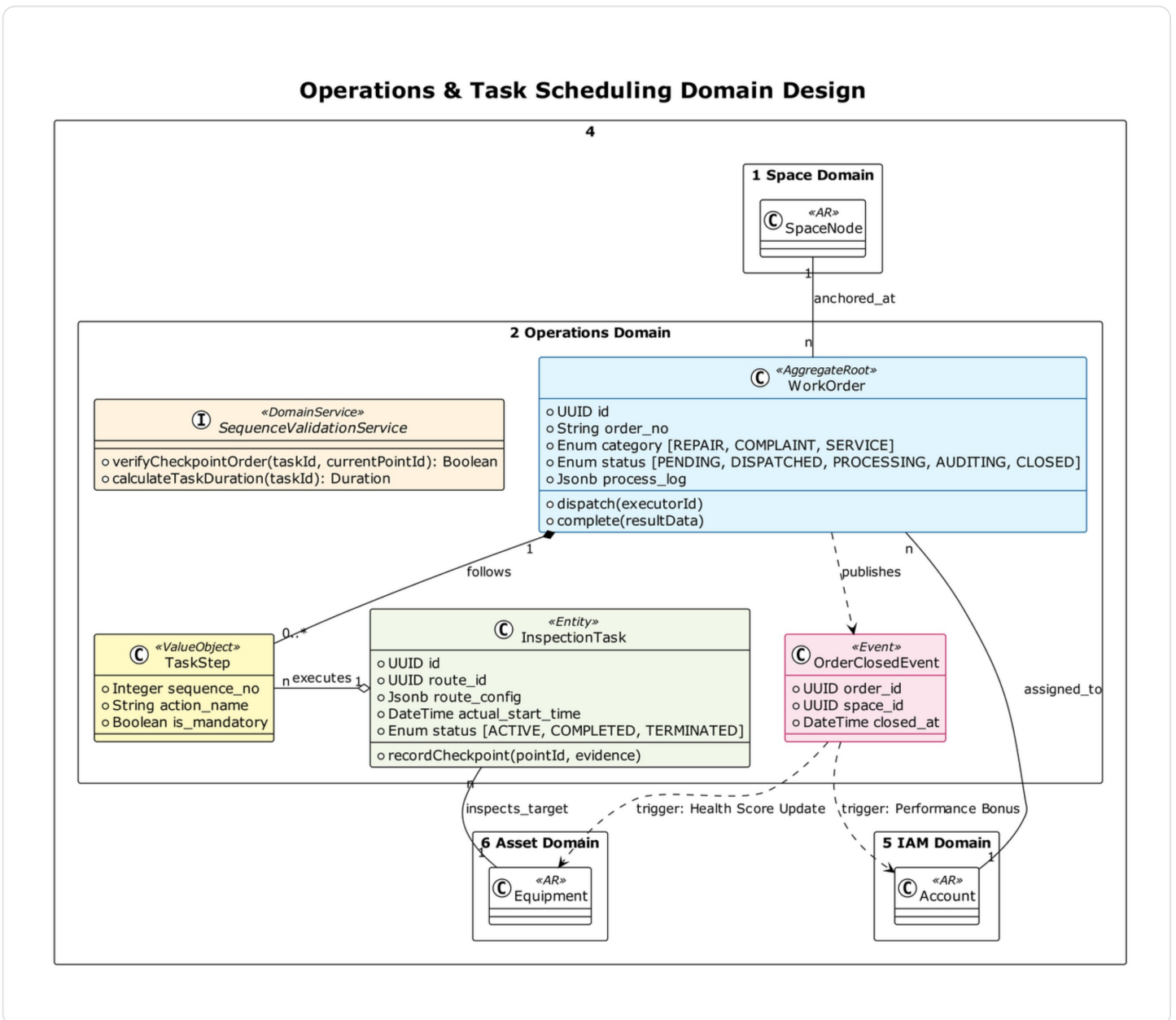
关键映射说明：

- **SpaceNode (空间节点)**: 核心聚合根，使用 `Ltree` 路径实现高性能树形检索。
- **PropertyBinding (房产绑定)**: 实现人房解耦，对应需规中业主、租户的多对多关系。
- **SpaceHierarchyService (空间层级服务)**: 跨聚合服务，处理节点移动和路径重组。

- **SpaceArchivedEvent (空间归档事件):** 当房间注销时抛出，触发财务域 (Finance Domain) 的账单自动清算。
- **External References:** 展示了设备 (Equipment) 和工单 (WorkOrder) 如何通过 `SpaceID` 锚定物理位置。

2. 运营调度领域DDD

UML图:



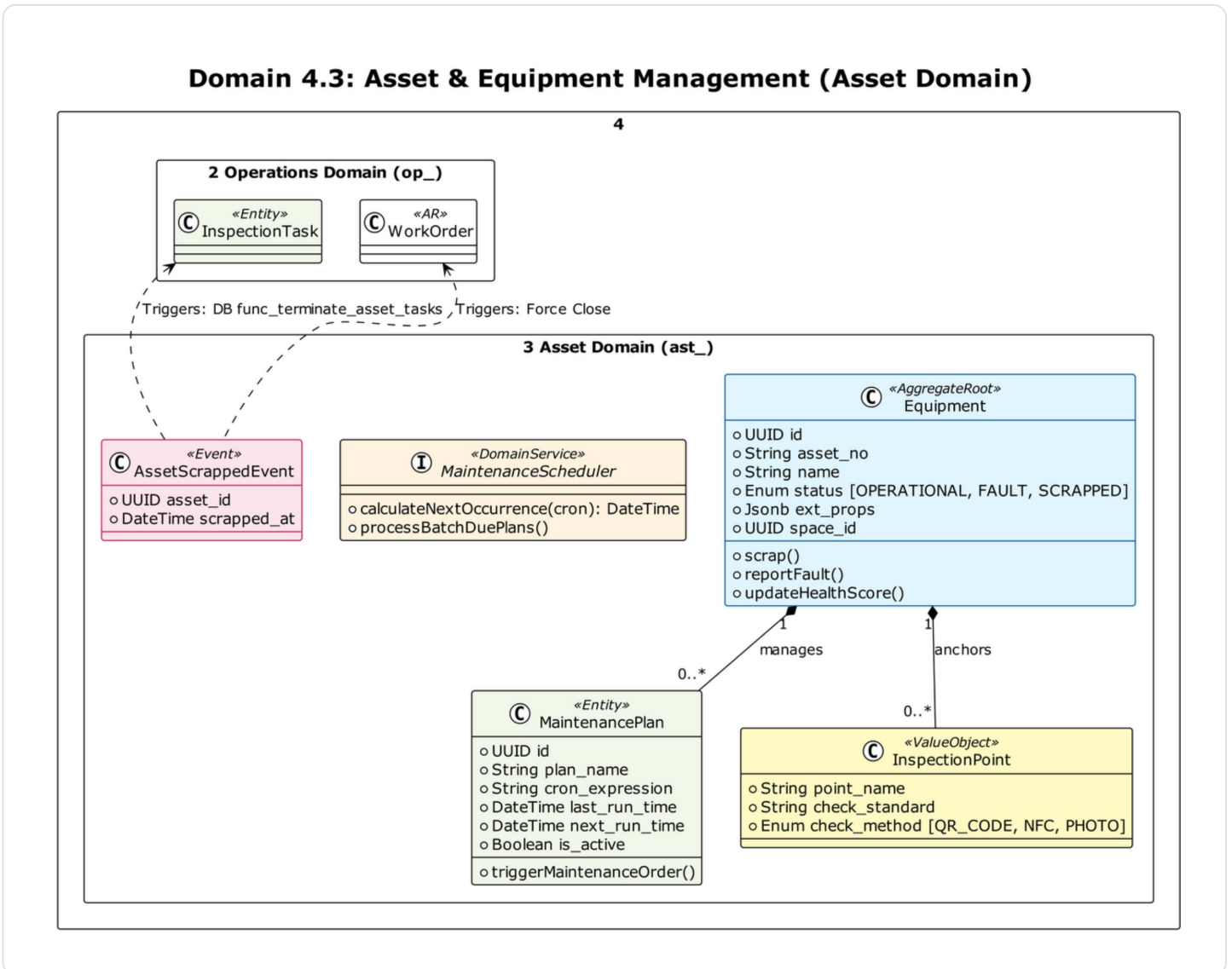
关键映射说明:

- **WorkOrder (综合工单聚合根):** 统一承载报修、投诉等业务。状态流转严格受限，通过 `process_log` (JSONB) 记录完整的审计追踪，满足需规中对过程留痕的要求。

- **SequenceValidationService (时序校验服务)**: 针对 **3.2.3 巡检模块** 的核心痛点。它负责比对 `route_config` 中的顺序与实际提交的 `recordCheckpoint` 顺序。如果顺序不一致，将触发预警或禁止提交，防止“假巡检”。
- **TaskStep (任务步骤)**: 作为值对象，定义了工单或巡检的标准化动作（如：拍照、签字、扫描二维码），确保作业过程的标准化。
- **领域事件联动: OrderClosedEvent (结单事件)**: 工单关闭后，系统自动通知 **4.6 设备领域** 更新对应设备的“健康得分”，或者通知 **4.5 权限领域** 为员工计算绩效积分，实现业务自动化闭环。
- **位置与资产关联**: 所有运营任务必须通过 `SpaceID` 或 `AssetID` 锚定。这保证了在 **4.1 空间领域** 查看某房间时，能反向检索出该位置发生过的所有历史报修记录。

3. 设施设备领域DDD

UML图:



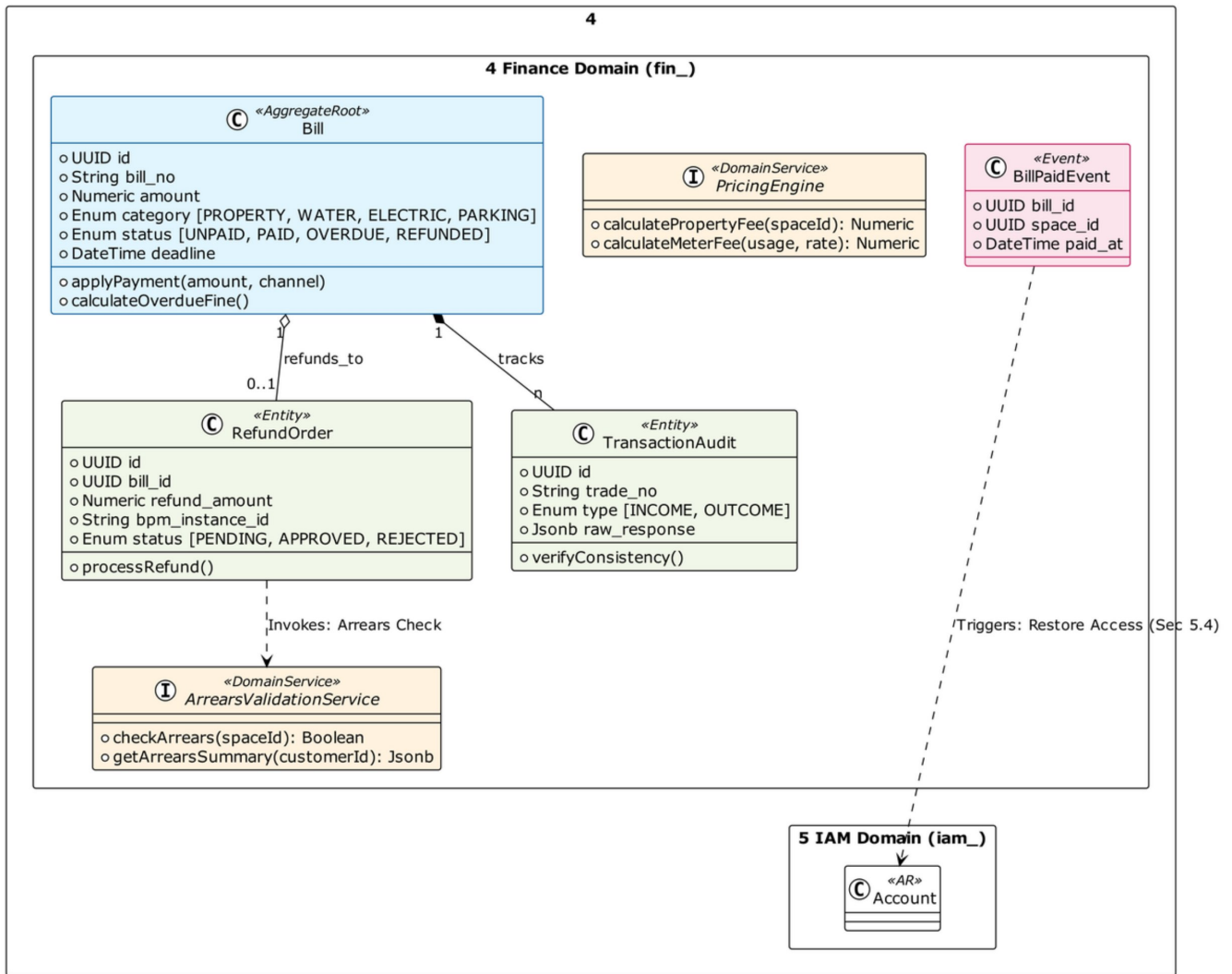
关键映射说明：

- **Equipment (设施设备聚合根)**：作为领域核心，使用 `ast_equipment` 表存储。通过 `ext_props` (JSONB) 兼容不同类型设备的特有属性（如电梯载重、水泵功率）。其 `status` 字段是状态机的核心，直接驱动后续的业务熔断逻辑。
 - **MaintenancePlan (维保计划实体)**：对应需规 3.2.2。通过 `cron_expression` 定义周期性维保频率。由领域服务 `MaintenanceScheduler` 定期扫描，自动调用 4.2 运营领域接口生成来源为“定时维保”的工单，实现从“被动维修”向“主动维保”的转型。
 - **InspectionPoint (巡检点位值对象)**：对应需规 3.2.3。定义了设备巡检时的标准化动作和校验介质（如 QR_CODE, NFC）。这些预设值是 4.2 运营领域执行巡检任务时进行“真实性校验”的唯一数据基准。
 - **AssetScrappedEvent (资产报废事件)**：对应技术方案 5.5。当设备执行 `scrap()` 操作且状态变更为 `SCRAPPED` 时抛出。该事件在数据库层面通过 `func_terminate_asset_tasks()` 触发器，强制清理所有 `op_` 开头的关联在办任务。
 - **空间与位置锚定**：通过 `space_id` 强关联 4.1 空间领域。确保所有资产均有明确的物理坐标（如：某项目-某楼栋-某机房），为 B 端管理后台提供基于地理位置的资产分布视图和故障热点分析。
-

4. 财务计费领域DDD

UML图：

Domain 4.4: Finance & Billing Management (Finance Domain)



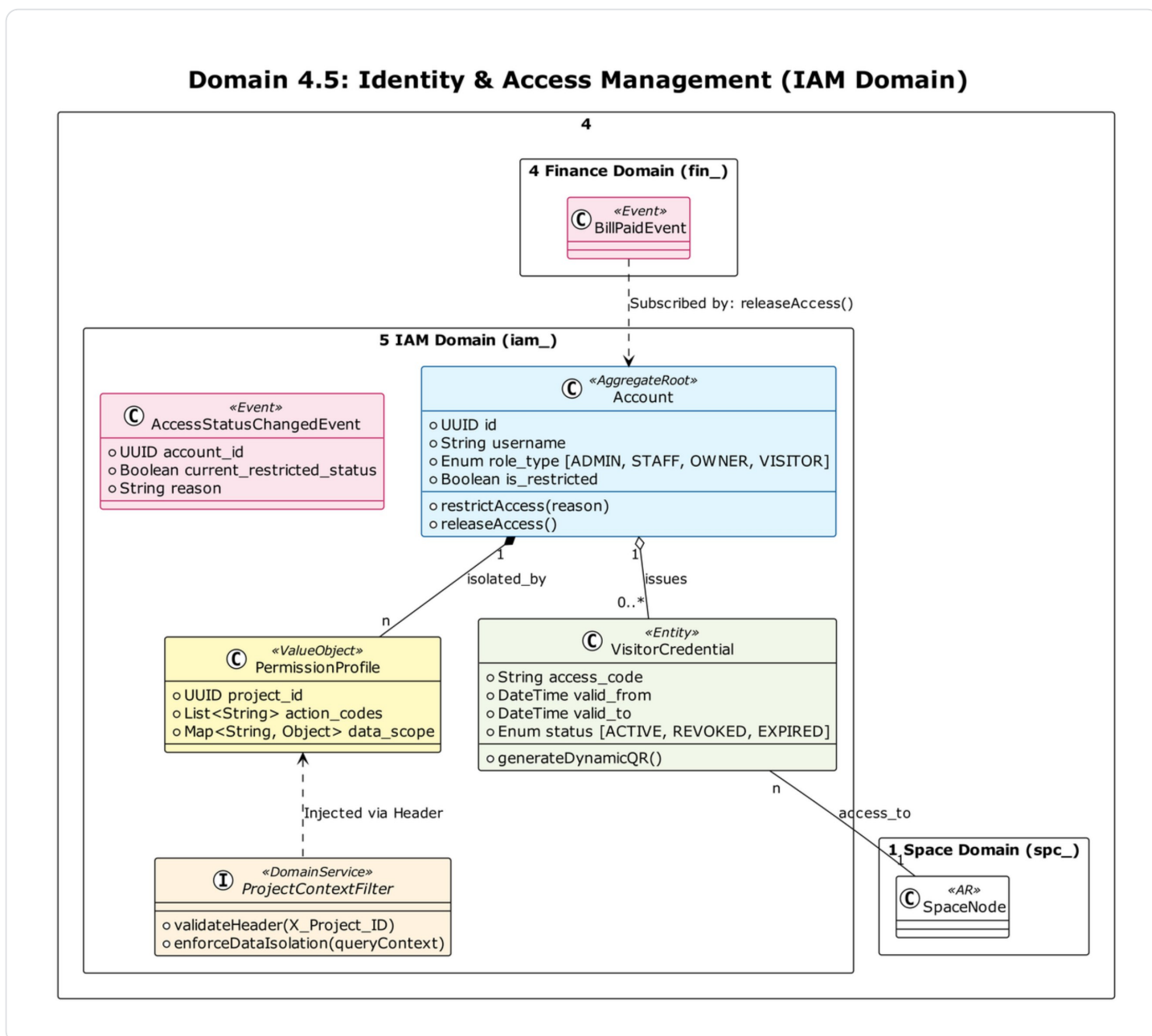
关键映射说明：

- **Bill (账单聚合根)**: 使用 `fin_bill` 表存储。统一承载各类周期性费用（物业费）与表计费用（水电费）。其 `status` 字段驱动整个财务生命周期，并利用 PostgreSQL 的 `Numeric` 类型确保金额计算的精度。
- **RefundOrder (退款单实体)**: 对应需规 3.5.2。作为退款业务的载体，关联 `bpm_instance_id` 以对接审批流。它通过调用领域服务执行“欠费对冲”，确保在处理退款前，系统已完成针对当前物理空间的所有债务核销。
- **PricingEngine (计费引擎服务)**: 该服务从 4.1 空间领域获取面积参数，从 4.6 集成中心获取表计读数，结合预设费率公式计算应收金额。它解耦了计费策略与底层存储，支持阶梯水价、滞纳金等复杂算法。
- **ArrearsValidationService (欠费校验服务)**: 针对 3.5.2 模块的核心风控逻辑。在退款操作前强制执行，扫描 `fin_bill` 中所有 `status = 'UNPAID'` 的记录。如果存在欠费，则阻断退款流程，体现了“先结清、再退款”的管理原则。

- **领域事件联动 (BillPaidEvent):** 对应技术方案 5.4。当账单状态转为 `PAID` 时抛出此事件。4.5 权限领域订阅后，自动解除对应 `Account` 的权限管控限制（如恢复门禁权限），实现“缴费-解锁”的自动化闭环。

5. 权限与账户领域DDD

UML图:



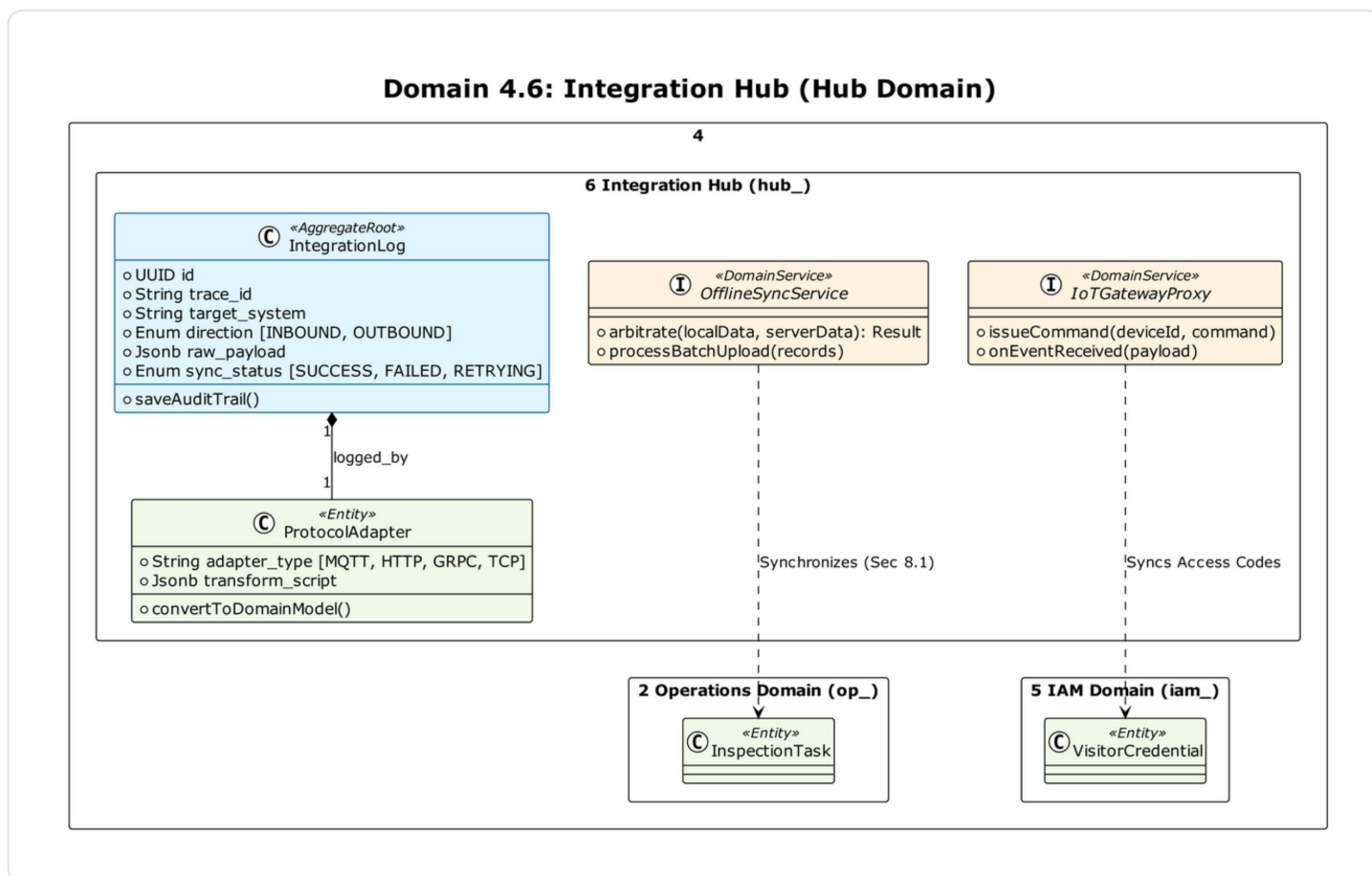
关键映射说明:

- **Account (账户聚合根):** 使用 `iam_account` 表存储，统一管理 B 端物业人员、C 端业主及临时访客。新增 `is_restricted` 标志位，用于承载由于欠费或其他违规导致的“权限熔断”状态，该状态将实时同步至 IoT 门禁系统。

- **ProjectContextFilter (项目上下文过滤服务)**: 严格对应技术规范 7.1。该服务作为系统拦截器，通过校验请求头的 `X-Project-ID`，实时从 `PermissionProfile` 中加载对应项目的权限空间，实现物理单体架构下的逻辑多租户深度隔离。
- **VisitorCredential (访客凭证实体)**: 对应需规 5.2。负责生成带有时效性的动态加密二维码或随机码。它与 4.1 空间领域锚定，确保访客仅能通过其预约空间所属的特定单元门禁，满足安全性要求。
- **PermissionProfile (权限配置值对象)**: 存储具体的 `action_codes` 和数据范围。它解决了文档中提出的“禁止从 Session 获取权限”的要求，所有权限判定必须基于当前活跃的项目上下文动态计算。
- **领域事件联动 (AccessStatusChangedEvent)**: 对应技术方案 5.4。当财务领域抛出 `BillPaidEvent` 后，本领域订阅并自动调用 `releaseAccess()`，将解除受限状态的消息通过 `iam_` 广播至集成中心，实现缴费后门禁自动恢复通行的业务闭环。

6. 集成中心领域DDD

UML图:



关键映射说明:

- **IntegrationLog (集成日志聚合根)**: 使用 `hub_integration_log` 表存储。该模型是异构系统对接的“黑匣子”，通过 `JSONB` 格式完整保存原始报文。它不仅用于审计，还支持在第三方系统故障恢复后的“指令重放 (Replay)”，确保跨系统交互的最终一致性。
- **OfflineSyncService (离线同步服务)**: 对应技术规范 8.1。针对移动端在地下室、电梯等弱网环境下产生的高频巡检数据，该服务实现了**时间戳仲裁 (Arbitration)** 逻辑。它负责在设备恢复网络后，对比本地记录与服务器状态，通过幂等校验解决数据冲突，防止重复或过时数据的覆盖。
- **ProtocolAdapter (协议适配器实体)**: 负责将外部系统（如停车场的 HTTP 接口或电梯监控的 MQTT 消息）转换为系统内部的领域模型。这种适配器模式将核心业务逻辑与外部协议细节彻底解耦，使得新增第三方集成时无需修改核心领域代码。
- **IoTGatewayProxy (IoT 网关代理)**: 作为硬件通信的桥梁。它负责将 4.5 权限领域生成的访客凭证 (VisitorCredential) 下发至物理门禁，并将 4.3 设施设备领域需要的 IoT 传感器实时数据转化为领域事件（如设备故障告警），触发后续工单流程。
- **审计与留痕**: 通过 `trace_id` 实现跨工程（`core` 与 `hub`）的全链路追踪。任何从外部系统进入的数据流，在集成中心都会被分配唯一的追踪标识，满足需规中对政府监管数据对接时“可追溯、不可篡改”的要求。

